# MultiWii NAV Protocol and Types

This document describes a number of values and enumerations for the stated version of the beta NAV development for MultiWii. This information is provided in the hope it might be useful **NO WARRANTY**.

Note that all binary values are little endian (MSP standard).

# MultiWii NAV Version

This document should match the 2.3-pre8 / b5 MultiWii / Wingui release.

2016-07-20. Update for iNav 1.2

### WayPoint / Action Attributes

Each waypoint has a type and takes a number of parameters, as below.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **TYPE** | **P1** | **P2** | **P3** | **Lat** | **Lon** | **Alt** |
| 1 | WAYPOINT | cm/s |  |  | ✔ | ✔ | ✔ |
| 2 | POSHOLD\_UNLIM |  |  |  | ✔ | ✔ | ✔ |
| 3 | POSHOLD TIME | Seconds |  |  | ✔ | ✔ | ✔ |
| 4 | RTH | Land |  |  |  |  | ✔ |
| 5 | SET POI |  |  |  | ✔ | ✔ |  |
| 6 | JUMP | WP# | Repeat (-1 = forever) |  |  |  |  |
| 7 | SET HEAD | Head |  |  |  |  |  |
| 8 | LAND |  |  |  | ✔ | ✔ | ✔ |

Note: Once SET\_HEAD is invoked, it remains active until cleared by a P1 value of -1. SET\_POI is also cleared by SET\_HEAD with P1 = -1.

Note: For JUMP, the target WP (P1) must be prior to the current WP.

Note: iNAV allows a leg speed to be set in WAYPOINT P1. The units are cm/sec.

Note: iNav 1.2 supports 1,2,4 above.

# Uploading

For safety, if no mission is defined, a single RTH action should be sent:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ACTION** | **P1** | **P2** | **P3** | **Lat** | **Lon** | **Alt** | **Flag** |
| RTH | 0 | 0 | 0 | 0 | 0 | 25m [0] | 0xa5 |

[0] your choice, really.

In general, flag is 0, unless it's the last point in a mission, in which case it is set to 0xa5. When waypoints are uploaded, the values are also returned by the FC, thus enabling the application to verify that the mission has been uploaded correctly.

# Capabilities

|  |  |
| --- | --- |
| **Capability** | **Value** |
| BIND | 1 |
| DYNBAL | 4 |
| FLAP | 8 |
| NAV | 16 |

Note that 32bit flight controllers (baseflight, cleanflight) use capability == 16 for a different purpose (CAP\_CHANNEL\_FORWARDING). Should cleanflight adopt MW semantics for WP navigation, this needs resolution.

Until that happens, it should be assumed that 32bit FCs are not nav capable.

# Messages (Nav related)

|  |  |  |
| --- | --- | --- |
| **MNEMONIC** | **Value** | **Direction (relative to FC)** |
| MSP\_NAV\_STATUS | 121 | Out |
| MSP\_NAV\_CONFIG | 122 | Out |
| MSP\_WP | 118 | Out |
| MSP\_RADIO | 199 | Out |
| MSP\_SET\_NAV\_CONFIG | 215 | In |
| MSP\_SET\_HEAD | 211 | In |
| MSP\_SET\_WP | 209 | In (& out) |

# MSP\_WP / MSP\_SET\_WP

Special waypoints are 0 and 255. 0 is the RTH position, 255 is the POSHOLD position (lat, lon, alt).

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Usage** |
| wp\_no | uchar | way point number |
| action | uchar | action (wp type / action) |
| lat | int32 | decimal degrees latitude \* 10,000,000 |
| lon | int32 | decimal degrees longitude \* 10,000,000 |
| altitude | int32 | altitude (metre) \* 100 |
| p1 | int16 | varies according to action |
| p2 | int16 | varies according to action |
| p3 | int16 | varies according to action |
| flag | uchar | 0xa5 = last, otherwise set to 0 |

The values for the various parameters are given in the section “WayPoint / Action Attributes”

# MSP\_NAV\_STATUS

The following data are returned by a MSP\_NAV\_STATUS message.

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Usage** |
| gps\_mode | uchar | "None",  "PosHold",  "RTH",  "Mission" |
| nav\_state | uchar | "None",  "RTH Start",  "RTH Enroute",  "PosHold infinit",  "PosHold timed",  "WP Enroute",  "Process next",  "Jump",  "Start Land",  "Land in Progress"  ,"Landed",  "Settling before land",  "Start descent" |
| action | uchar | (last wp, next wp?) |
| wp\_number | uchar | (last wp, next wp?) |
| nav\_error | uchar | "Navigation system is working.",  ”Next waypoint distance is more than the safety limit, aborting mission",  "GPS reception is compromised - pausing mission, COPTER IS ADRIFT!",  "Error while reading next waypoint from memory, aborting mission.",  "Mission Finished." ,  "Waiting for timed position hold.",  "Invalid Jump target detected, aborting mission.",  "Invalid Mission Step Action code detected, aborting mission.",  "Waiting to reach return to home altitude.",  "GPS fix lost, mission aborted - COPTER IS ADRIFT!",  "Copter is disarmed, navigation engine disabled.",  "Landing is in progress, check attitude if possible." |
| target\_bearing | int16 | (presumably to the next WP?) |

# MSP\_NAV\_CONFIG

The following data are returned from a MSP\_NAV\_CONFIG message. Values from config.h. Values may also be set by MSP\_SET\_NAV\_CONFIG.

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Usage** |
| flags1 | uchar | Bitmap of settings from config.h  b0 : GPS filtering  b1 : GPS Lead  b2 : Reset Home  b3 : Heading control  b4 : Tail first  b5 : RTH Head  b6 : Slow Nav  b7 : RTH Alt |
| flags2 | uchar | Bitmap of settings from config.h  b0 : Disable sticks  b1 : Baro takeover |
| wp\_radius | uint16 | radius around which waypoint is reached  (cm) |
| safe\_wp\_distance | uint16 | Maximum permitted first leg of mission  (m, assumed?) |
| nav\_max\_altitude | uint16 | Maximum altitude for NAV  (m) |
| nav\_speed\_max | uint16 | maximum speed for NAV  cm/sec |
| nav\_speed\_min | uint16 | minimum speed for NAV  (cm/s) |
| crosstrack\_gain | uchar | config.h value \* 100 |
| nav\_bank\_max | uint16 | maximum bank ??? for NAV  config.h value \*100 |
| rth\_altitude | uint16 | RTH altitude  ( m) |
| land\_speed | uchar | Governs the descent speed during landing. 100 ~= 50 cm/sec  unknown units |
| fence | uint16 | Distance beyond which forces RTH  (m) |
| max\_wp\_number | uchar | maximum number of waypoints possible (read only) |

# MSP\_RADIO

If you have a 3DR radio with the MW/MSP specific firmware, the follow data are sent from the radio, unsolicited.

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Usage** |
| rxerrors | uint16 | Number of RX errors |
| fixed\_errors | uint16 | Number of fixed errors, if error correction is set |
| localrssi | uchar | Local RSSI |
| remrssi | uchar | Remote RSSI |
| txbuf | uchar | Size of TX buffer |
| noise | uchar | Local noise |
| remnoise | uchar | Remote noise |

The MSP NAV message set is implemented by **mwptools** (Linux), **ezgui** (Android) and **WinGUI** (MS Windows).

written by Jonathan Hudson <[jh+mwp@daria.co.uk](mailto:jh+mwp@daria.co.uk)>